Visual Information Processing

Menandro Roxas
Computer Vision Laboratory
October 31, 2018
Topic

- Spatio-Temporal Image Processing / Video Processing
  - 10/31 – What kind of information can we gather from an image sequence?
  - 11/07 – In what applications can we use this information?
Image vs. Image Sequence

• Low Level Information
  • Image
    • Color (RGB, Intensity)
    • Gradient

• Image Sequence
  • Pixel motion (optical flow)

• High Level Information (inferred)
  • Image
    • Object classification

• Image Sequence
  • Object motion
  • 3D structure
  • Camera pose
# Image vs. Image Sequence

<table>
<thead>
<tr>
<th>Low level information</th>
<th>High level information</th>
</tr>
</thead>
<tbody>
<tr>
<td>color, gradient</td>
<td>object classification</td>
</tr>
<tr>
<td><img src="image.png" alt="image" /></td>
<td><img src="object.png" alt="object classification" /></td>
</tr>
<tr>
<td><img src="image_sequence.png" alt="image sequence" /></td>
<td><img src="object_motion.png" alt="object motion" /></td>
</tr>
<tr>
<td>pixel motion (optical flow)</td>
<td>3D structure</td>
</tr>
<tr>
<td><img src="pixel_motion.png" alt="pixel motion" /></td>
<td>camera pose</td>
</tr>
<tr>
<td><img src="image_sequence.png" alt="image sequence" /></td>
<td><img src="camera_pose.png" alt="camera pose" /></td>
</tr>
</tbody>
</table>
Spatio-Temporal Image Processing

• Observation of pixel change over time
  • Is it changing color?
  • Is it moving?

• What can we infer from these changes?
  • Distance of the object
  • Speed
  • Camera motion
Application (Overview)

• Video Inpainting
Application (Overview)

• 3D Reconstruction (BigSfM: Reconstructing the World from internet photos)
Optical Flow

Apparent motion of objects
Optical Flow
Optical Flow

- **Brightness Constancy**

\[ I(x + u, y + v, t + 1) - I(x, y, t) = 0 \]

- **First-order approximation**

\[ I(x + u, y + v, t + 1) \approx I(x, y, t) + \frac{dl}{dx} u + \frac{dl}{dy} v + \frac{dl}{dt} \]

\[ I(x + u, y + v, t + 1) - I(x, y, t) = \frac{dl}{dx} u + \frac{dl}{dy} v + \frac{dl}{dt} = I_x u + I_y v + I_t \]
Optical Flow (1D)

\[ I_x = \frac{\delta I}{\delta x} \bigg|_t \quad I_t = \frac{\delta I}{\delta t} \bigg|_{x=p} \quad \vec{u} \approx -\frac{I_t}{I_x} \]

Assumptions:
- brightness constancy
- small motion
Optical Flow

• How to solve \((u, v)\)?

\[ I_x u + I_y v + I_t = 0 \]

• Discretize in image domain

\[ I_x = I(x + 1, y) - I(x, y) \]
\[ I_y = I(x, y + 1) - I(x, y) \]
\[ I_t = I(x, y, t) - I(x, y, t + 1) \]

• One equation (scalar), two unknowns \((u, v)\):
Optical Flow

• How to solve \((u, v)\)?
  \[ I_x u + I_y v + I_t = 0 \]

• Discretize in image domain
  \[ I_x = I(x + 1, y) - I(x, y) \]
  \[ I_y = I(x, y + 1) - I(x, y) \]
  \[ I_t = I(x, y, t) - I(x, y, t + 1) \]

• One equation (scalar), two unknowns \((u, v)\)?
Optical Flow
(Simple solution)

• Over-constrained
  • Assume: neighboring pixels have equal motion

• More equations

\[
\begin{align*}
I_x u + I_y v &= I_t \\
\bar{I}_x u + \bar{I}_y v &= \bar{I}_t \\
\begin{bmatrix}
I_x & I_y \\
\bar{I}_x & \bar{I}_y 
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix}
&= 
\begin{bmatrix}
-I_t \\
-\bar{I}_t
\end{bmatrix}
\end{align*}
\]

• Problem:
  • Works only for single object
Issues with Optical Flow

• Aperture problem
• Occlusions
• Textureless Region
• Large Motion
## Issues with Optical Flow

<table>
<thead>
<tr>
<th>Aperture problem</th>
<th>Occlusion</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Aperture problem" /></td>
<td><img src="image2" alt="Occlusion" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Textureless region</th>
<th>Large motion</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image3" alt="Textureless region" /></td>
<td><img src="image4" alt="Large motion" /></td>
</tr>
</tbody>
</table>
Apperture Problem: Solution

- Isotropic information
  - Corners
  - High textures
  - Not always present

- Regularization
  - Smoothness constraint
  - Neighboring pixels have similar motion (more or less)
Apperture Problem: Solution

• Use isotropic information
  • Corners
  • High textures
Aperture Problem: Solution

• Neighboring pixels have “similar” motion

\[
\begin{align*}
  u(x, y) & \quad u(x+1, y) \\
  u(x, y+1) & \quad \frac{du}{dx} \approx 0 \\
  & \quad u(x + 1, y) - u(x, y) \approx 0
\end{align*}
\]

• Smoothness Constraint

\[
\arg \min_{u,v} \left( \frac{du}{dx} \right)^2 + \left( \frac{du}{dy} \right)^2 + \left( \frac{dv}{dx} \right)^2 + \left( \frac{dv}{dy} \right)^2
\]
Occlusion

• Undefined motion in occluded area
  • But still constrained. How?
Occlusion: Solution

• Step 1: Detection of occluded area

Frame 1  Frame 2  Optical Flow

• Step 2: Deciding what “kind” of motion it is

or

GUESS based on the color!
Textureless Region

- Image gradients are zero

\[ I_x = 0 \quad I_y = 0 \]
Textureless Region

Local neighborhood

Regularization

Textured information

Too far!!
Textureless Regions: Solution

- Coarse-to-fine approach + Regularization
  - Make the image small
  - Make neighboring pixels have similar motion
  - Increase resolution
Large Motion

Must be this!
Solved $I_t$

Actual corresponding points
Large Motion: Solution

- Coarse-to-fine approach
  - Bring the two corresponding points closer
Large Motion: Solution

- Sparse matching (global scale feature matching)
  - Search whole image for similar features
Methods for Solving Optical Flow

• Feature-matching method
  – Patch matching
  – Naturally sparse and discrete

• Variational method
  – Naturally dense (complete) and continuous
Feature Matching Methods

- Handles large displacement
- Mostly sparse

- SIFT, SURF, SIFT-Flow etc.
- DeepMatching

- A topic for a different day...
Variational Method

- Minimize the sum of convex energies
  - Data (Brightness constancy)
  - Regularizer (Smoothness constraint)
  - Large Displacement Handling
  - Epipolar Geometry

$$\arg \min_{u,v} F_{data}(u, v) + F_{reg}(u, v) + F_{ld}(u, v) \ldots$$
Variational Method

• Brightness Constancy
  • Lucas-Kanade
    \[ I_x u + I_y v + I_t = 0 \]

• Regularization
  • Least-Squares (Horn-Schunk)
    \[ \left( \frac{du}{dx} \right)^2 + \left( \frac{du}{dy} \right)^2 = 0 \]
  • Total Variation
    \[ \sqrt{\left( \frac{du}{dx} \right)^2 + \left( \frac{du}{dy} \right)^2} = 0 \]
  • Non-Local (Median Filter + extension)
Regularization

• Total variation (TV) \[ \sqrt{\left(\frac{du}{dx}\right)^2 + \left(\frac{du}{dy}\right)^2} \]

![Diagram of signal and denoising process](image)
Regularization

• Median filtering vs. Weighted Median filtering
Regularization

• Comparison of regularizers

(a) “Old” HS [58]  (b) “New” HS  (c) Classic++  (d) Classic+NL  (e) Ground truth  (f) First frame

Least-squares  Total Variation  w/ median filtering  w/ weighted median filtering
Handling Large Displacement

- Global feature matching
- Constrain the pixel to the detected motion

\[ u(x, y) - \bar{u}(x, y) = 0 \]
Special Case: Epipolar Geometry

- Constrain the motion along a trajectory
  - Only for moving camera and stationary objects
Special Case: Epipolar Geometry

Camera position 0

Camera position 1

SCENE

Fundamental Matrix, F

\[ x_1^T F x_0 = 0 \]
\[ (x_0 + u)^T F x_0 = 0 \]

\( x_1 \) and \( x_0 \) are corresponding points (i.e. \( x_1 = x_0 + u \))
Example of Minimization Function

\[
\arg \min_{u,v} \sqrt{(I_x u + I_y v + I_t)^2 + \varepsilon^2 + \lambda_{TV} \sqrt{(\frac{du}{dx})^2 + (\frac{du}{dy})^2} + \lambda_{TV} \sqrt{(\frac{dv}{dx})^2 + (\frac{dv}{dy})^2}} \\
+ \lambda_{median} \phi_{median}(u) + \lambda_{median} \phi_{median}(v) \\
+ \lambda_{matching} (u - u_{matching})^2 + \lambda_{matching} (v - v_{matching})^2 \\
+ ((x_0 + u)^T F x_0)^2
\]
Solution to Variational Method

• Euler-Lagrange
• Gradient descent
• Primal-Dual decomposition
  • Used for non-differentiable functions such as l1-norm
Evaluation Databases

vision.middlebury.edu
stereo • mview • MRF • flow • color

MPI Sintel Flow Dataset
A dataset for the evaluation of optical flow derived from the open source 3D animated short film, Sintel.
Evaluation Database: Middlebury

• 12 training sets
• 12 testing sets

• Advantages
  – Real + synthetic images
  – Random object motion

• Disadvantages
  – Very small motion (~20 pixels)
  – Few images
Evaluation Database: KITTI2012

• 193 traning
• 194 testing

• Advantages
  – Real world images

• Disadvantages
  – Partial ground truth
  – Outdoor only
  – Motions are due to camera movement (Static scene)
Evaluation Database : KITTI2015

• 200 training
• 200 testing

• Advantages
  – Real world images
  – Dynamic motion (moving objects)
  – More challenging

• Disadvantages
  – Challenging sets are very few (difficult to make conclusive remarks)
Evaluation Database: MPI Sintel

• 23 training
• 12 testing

• Advantages
  – Dynamic motion
  – Challenging large motion
  – Most difficult of the three database (Middlebury, Kitti and MPI)

• Disadvantages
  – Synthetic (CGI)
Best Performing Methods

• General Purpose
  • FlowFields
    • C. Bailer et al., “Flow Fields: Dense Correspondence Fields for Highly Accurate Large Displacement Optical Flow Estimation”, ICCV 2015
  • EpicFlow

• Epipolar Geometry
  • PRSM
    • C. Vogel et al., “3D Scene Flow Estimation with a Piecewise Rigid Scene Model”, IJCV 2015
  • SPS-FI

• Variational
  • Classic++, Classic++NL

• Variational ++
  • DeepFlow

• Learning-Based (CNN)
  • FlowNet2
FlowFields

For $k=2$
$n=4=2^2$
$n=2$
$n=1$

KD-Tree initialization
Propagation:
Resolution increase

a) 

b) 
c) 
d) 
e) 
f) 

Random Search

3x
after 4th propagation

Full resolution?

yes

Done

no

Increase Resolution

KD-Tree Initialization -> Propagation -> Random Search

Increase Resolution -> no -> Full resolution? -> no

Full resolution? -> yes -> Done

Increase Resolution -> yes -> Full resolution? -> yes

Full resolution? -> yes -> Done
Results

(a) ANNF [16]  (b) Our Flow Field

(c) Our outlier filtered Flow Field  (d) Ground truth
EpicFlow

\[ F_{NW}(p) = \sum_{(p_m, p'_m) \in \mathcal{M}} \frac{k_D(p_m, p)p'_m}{\sum_{(p_m, p'_m) \in \mathcal{M}} k_D(p_m, p)} \]

Dense Interpolation

Figure 1. Image edges detected with SED [15] and ground-truth optical flow. Motion discontinuities appear most of the time at image edges.
Results
PRSM

- Piecewise rigid moving planes
SPS-Fi - Yamaguchi

- MRF model
  - Piecewise rigid moving planes (same as PRSM)

\[
\hat{d}(p, \theta_i) = A_ip_x + B_ip_y + C_i
\]

Disparity at pixel \( p \)

A, B, C are plane parameters

(px, py) is the center of a segment
Results
Classic++, Classic+NL

- Classic++, 5x5 median filter
- Classic+NL, Edge-adaptive median filter
  - Improved boundaries
DeepFlow

- Accurate, but DeepMatching is very slow
Results

Average frames

Deep Matching

DeepFlow

Groundtruth flow
FlowNet2

• CNN-based
• Cascaded network
Result

FlowNet 2.0 vs FlowFields

FlowNet 2.0 generates sharper boundaries, achieves comparable error scores, and runs ca. **200x faster**